UNITED STATES PATENT AND TRADEMARK OFFICE

IFW

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/728,551	12/04/2003	Christopher E. Phillips	42P16936C	8778

7590 08/09/2007

Blakely, Sokoloff, Taylor & Zafman LLP
Suite 101
5285 S.W. Meadows Road
Lake Oswego, OR 97035

EXAMINER	
CONNOLLY, MARK A	

ART UNIT	PAPER NUMBER
2115	

MAIL DATE	DELIVERY MODE
08/09/2007	PAPER

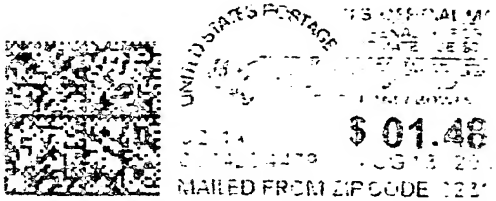
Please find below and/or attached an Office communication concerning this application or proceeding.

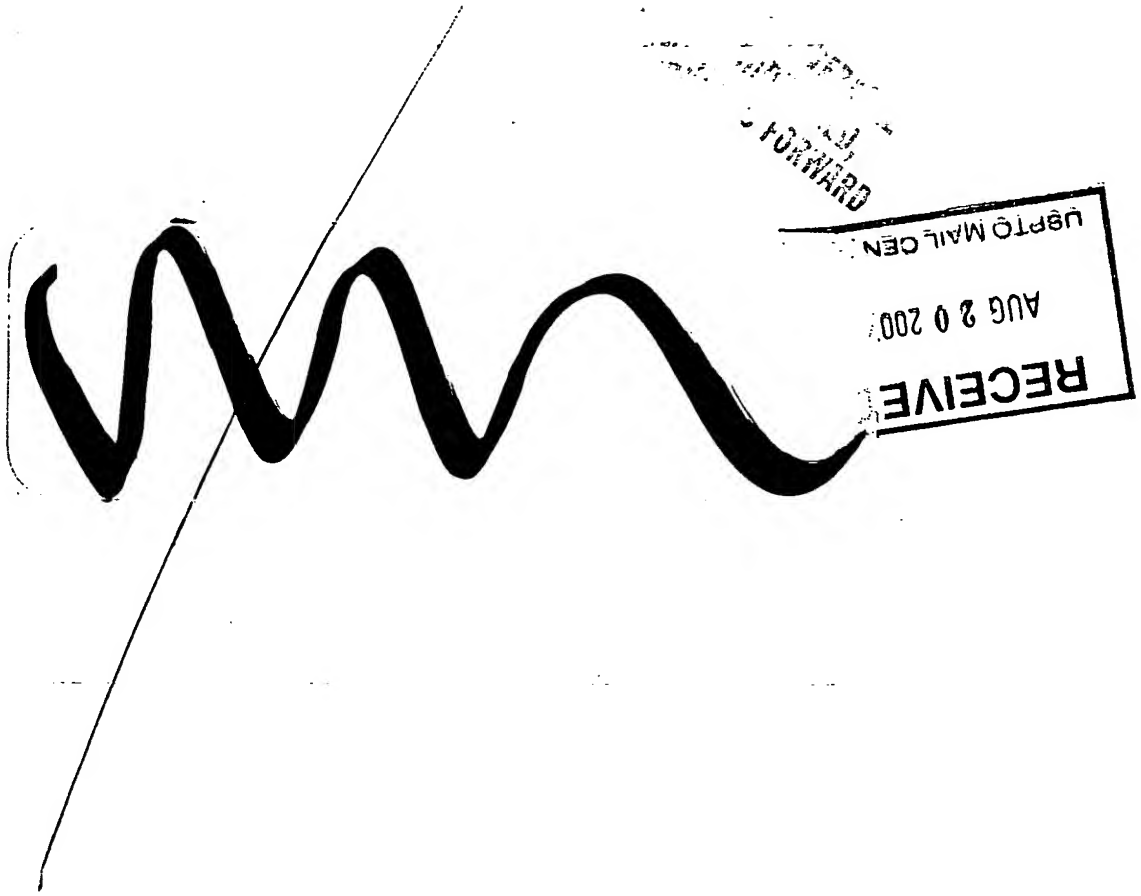
The time period for reply, if any, is set in the attached communication.

Organization_____ Bldg/Room_____
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
If Undeliverable Return in Ten Days

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

AN EQUAL OPPORTUNITY EMPLOYER





Office Action Summary	Application No.	Applicant(s)	
	10/728,551	PHILLIPS ET AL.	
	Examiner	Art Unit	
	Mark Connolly	2115	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 December 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *8-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *8,10-14,16-20 and 22* is/are rejected.

7)☒ Claim(s) *9,15 and 21* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *04 December 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☒ Other: *See Continuation Sheet*.

Continuation of Attachment(s) 6). Other: Revised Amendment Practce document.

DETAILED ACTION

1. Claims 8-22 have been presented for examination.

2. It should be initially noted that in the preliminary amendment filed 4 December 2003, applicant has included Claim Amendments and Remarks on the same page. Section I of the REVISED AMENDMENT PRACTICE (37 CFR 1.121) clearly states that "Each section of an amendment document (*e.g.*, Specification Amendments, Claim Amendments, Drawing Amendments, and Remarks) must begin on a separate sheet." Please see the attached flyer for further information regarding revised 37 CFR 1.121.

In the interest of compact prosecution the claims have been treated on the merits and which can be found below.

Specification

3. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required:

In particular, claim 1 explicitly teaches that a table *consists* of multiple entries, each entry identifying an on-chip configuration plane and a unique off-chip address. The specification rather teaches a cache with multiple cache lines (assumed by the examiner to be the table in claim 1) where each line *consists* of a configuration field (assumed as identifying an on-chip configuration plane), contents addressable field (i.e. off-chip address) and a *tagged bit field* [page 6 lines 8-12].

Claim Rejections - 35 USC § 112

4. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

5. Claim 1 is rejected under 35 U.S.C. 112, first paragraph, as based on a disclosure which is not enabling. The tagged bit field, which is "set to TRUE for any line with an address field that exactly matches a searched for address" is critical or essential to the practice of the invention, but not included in the claim(s) is not enabled by the disclosure. See *In re Mayhew*, 527 F.2d 1229, 188 USPQ 356 (CCPA 1976). The tagged bit field is required while searching the cache (i.e. table) for a given configuration [page 6 lines 12-14].

For examination purposes, claim 1 has been interpreted as having a table *comprising* multiple entries including an on-chip configuration plane and a unique off-chip address of a loaded configuration.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors Protection Act of 1999 (AIPA) and the Intellectual Property and High Technology Technical Amendments Act of 2002 do not apply when the reference is a U.S. patent resulting directly or indirectly from an international application filed before November 29, 2000. Therefore, the prior art date of the

reference is determined under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C. 102(e)).

7. Claim 8,10-14 and 16-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Vorbach US Pat No 6571381.

8. Referring to claim 8, Vorbach teaches the reconfiguration system including a reconfigurable computing chip having reconfigurable logic, multiple planes and off-chip storage performing the method comprising:

a. storing in a table a current state of the on-chip configurations [col. 3 lines 1-7, col. 5 lines 61-64 and col. 6 lines 18-21].

b. the table comprising multiple entries, each entry identifying an on-chip configuration plane and a unique off-chip address of a loaded configuration [col. 5 lines 61-64, col. 9 lines 23-24 and col. 11 lines 43-46].

In summary, Vorbach teaches a configuration table (CT) arranged in a tree structure, which, at the lowest levels (leaves) represent configurations for the individual configurable elements (CELs) in the reconfigurable system. The levels above the leaves store additional configurations for the CELs in a memory (CTR) and when a new configuration is requested, the LOAD request traverses up the tree structure requesting a configuration ID (KR<ID> which is the address of the configuration in external configuration memory (ECR)) until the required configuration is found wherein the configuration is then sent down to the leaf/leaves making the request.

9. Referring to claim 10, Vorbach teaches specifying an address of a boot configuration in the table and loading the boot configuration on boot up [col. 10 lines 52-55 and col. 11 lines 36-41].

10. Referring to claim 11, this is rejected on the same basis as set forth hereinabove.

11. Referring to claims 12 and 13, Vorbach teaches loading new configurations for reconfiguring the CELs within a reconfigurable system [col. 8 lines 42-45 and col. 9 lines 23-24]. The CELs are reconfigured if they are in a reconfigurable state. Reconfiguring the CELs is interpreted as activating a configuration plane with the loaded configuration. Since the old configuration is no longer loaded in the CELs, it is interpreted that they are unloaded from the active configuration plane.

12. Referring to claim 14, this is rejected on the same basis as set forth hereinabove.

13. Referring to claims 16 and 17, these are rejected on the same basis as set forth hereinabove. In particular, when a RESET occurs thus necessitating a boot sequence, the CT loads a boot configuration at address BOOT ADR, wherein the address is stored in the CT [col. 10 lines 52-55].

14. Referring to claim 18, this is rejected on the same basis as set forth hereinabove. Vorbach teaches the method and therefore teaches the system performing the method. In addition, Vorbach teaches having CTs comprising CTRs which cache configurations in addition to the ECR which stores all configurations. The CTs are interpreted as storing at least a subset of the configurations.

15. Referring to claim 19, Vorbach teaches the memory system is one or more memory devices [col. 5 lines 58-67].

16. Referring to claim 20, Vorbach teaches the memory system is a local internal memory [col. 5 lines 66-67].

Claim Rejections - 35 USC § 103

17. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

18. Claim 22 is rejected under 35 U.S.C. 103(a) as being unpatentable over Vorbach as applied to claims 8,10-14 and 16-20 above.

19. Referring to claim 22, Vorbach teaches a program pointer (PP) which points to configuration words (KW) within the configuration data to be executed [col. 8 lines 40-41]. Although Vorbach teaches that the KW being executed is in the CTS which are in the CTs, when booting, the all configuration data is located in the ECR memory. Since the PP is necessary to keep track of which KW should be executed, it is obvious that during boot, the PP would point to address locations in the ECR since that is where the KWs in the boot configuration are initially stored.

Allowable Subject Matter

20. Claims 9, 15 and 21 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

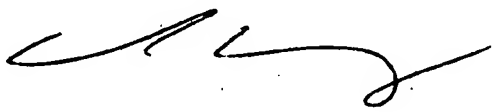
21. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mark Connolly whose telephone number is (571) 272-3666. The examiner can normally be reached on M-F 8AM-5PM (except every first Friday).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Thomas C. Lee can be reached on (571) 272-3667. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Mark Connolly
Examiner
Art Unit 2115

mc
December 22, 2006



REVISED AMENDMENT PRACTICE: 37 CFR 1.121 CHANGED COMPLIANCE IS MANDATORY - Effective Date: July 30, 2003

All amendments filed on or after the effective date noted above must comply with revised 37 CFR 1.121. See Final Rule: **Changes To Implement Electronic Maintenance of Official Patent Application Records** (68 *Fed. Reg.* 38611 (June 30, 2003), posted on the Office's website at: http://www.uspto.gov/web/patents/ifw/ with related information. The amendment practice set forth in revised 37 CFR 1.121, and described below, replaces the voluntary revised amendment format available to applicants since February 2003. **NOTE: STRICT COMPLIANCE WITH THE REVISED 37 CFR 1.121 IS REQUIRED AS OF THE EFFECTIVE DATE (July 30, 2003).** The Office will notify applicants of amendments that are not accepted because they do not comply with revised 37 CFR 1.121 via a Notice of Non-Compliant Amendment. See MPEP 714.03 (Rev. 1, Feb. 2003). The non-compliant section(s) will have to be corrected and the entire corrected section(s) resubmitted within a set period.

Bold underlined italic font has been used below to highlight the major differences between the revised 37 CFR 1.121 and the voluntary revised amendment format that applicants could use since February, 2003.
Note: The amendment practice for **reissues** and **reexamination proceedings**, except for drawings, has not changed.

REVISED AMENDMENT PRACTICE

I. Begin each section of an amendment document on a separate sheet:
Each section of an amendment document (*e.g.*, Specification Amendments, Claim Amendments, Drawing Amendments, and Remarks) must begin on a separate sheet. Starting each separate section on a new page will facilitate the process of separately indexing and scanning each section of an amendment document for placement in an image file wrapper.

II. Two versions of amended part(s) no longer required:
37 CFR 1.121 has been revised to no longer require two versions (a clean version and a marked up version) of each replacement paragraph or section, or amended claim. Note, however, the requirements for a clean version and a marked up version for substitute specifications under 37 CFR 1.125 have been retained.

A) Amendments to the claims:
Each amendment document that includes a change to an existing claim, cancellation of a claim or submission of a new claim, **must include a complete listing** of all claims in the application. After each claim number in the listing, the status must be indicated in a parenthetical expression, and **the text of each pending claim** (with markings to show current changes) must be presented. The claims in the listing will replace all prior claims in the application.

(1) The current status of all of the claims in the application, including any previously canceled, not entered or withdrawn claims, must be given in a parenthetical expression following the claim number using only one of the following seven status identifiers: (original), (currently amended), (canceled), (withdrawn), (new), ***(previously presented) and (not entered).*** The text of all pending claims, ***including withdrawn claims,*** must be submitted each time any claim is amended. Canceled ***and not entered*** claims must be indicated by only the claim number and status, without presenting the text of the claims.

(2) The text of all claims being currently amended must be presented in the claim listing with markings to indicate the changes that have been made relative to the immediate prior version. The changes in any amended claim must be shown by underlining (for added matter) or strikethrough (for deleted matter) with 2 exceptions: (1) for ***deletion of five characters or fewer, double brackets may be used (e.g., [[eroor]]); and (2) if strikethrough cannot be easily perceived (e.g., deletion of the number "4" or certain punctuation marks), double brackets must be used (e.g., [[4]]). As an alternative to using double brackets, however, extra portions of text may be included before and after text being deleted, all in strikethrough, followed by including and underlining the extra text with the desired change*** *(e.g.,* ~~*number 4 as*~~ *number 14 as).* An accompanying clean version is not required and should not be presented. Only claims of the status "currently amended," and "withdrawn" that are being amended, may include markings.

(3) The text of pending claims not being currently amended, ***including withdrawn claims,*** must be presented in the claim listing in clean version, *i.e.*, without any markings. Any claim text presented in clean version will constitute an assertion that it has not been changed relative to the immediate prior version except to omit markings that may have been present in the immediate prior version of the claims.

(4) A claim being canceled must be listed in the claim listing with the status identifier "canceled"; the text of the claim must not be presented. Providing an instruction to cancel is optional.
(5) Any claims added by amendment must be presented in the claim listing with the status identifier "(new)"; the text of the claim must not be underlined.
(6) All of the claims in the claim listing must be presented in ascending numerical order. Consecutive canceled, or not entered, claims may be aggregated into one statement (*e.g.*, Claims 1 – 5 (canceled)).

Example of listing of claims (use of the word "claim" before the claim number is optional):

Claims 1-5 (canceled)

Claim 6 (previously presented): A bucket with a handle.

Claim 7 (withdrawn): A handle comprising an elongated wire.

Claim 8 (withdrawn): The handle of claim 7 further comprising a plastic grip.

Claim 9 (currently amended): A bucket with a ~~green~~ blue handle.

Claim 10 (original): The bucket of claim 9 wherein the handle is made of wood.

Claim 11 (canceled)

Claim 12 (not entered)

Claim 13 (new): A bucket with plastic sides and bottom.

B) Amendments to the specification:
Amendments to the specification, including the abstract, must be made by presenting a replacement paragraph or section or abstract marked up to show changes made relative to the immediate prior version. An accompanying clean version is not required and should not be presented. Newly added paragraphs or sections, including a new abstract (instead of a replacement abstract), must not be underlined. A replacement or new abstract must be submitted on a separate sheet, 37 CFR 1.72. If a substitute specification is being submitted to incorporate extensive amendments, both a clean version (which will be entered) and a marked up version must be submitted as per 37 CFR 1.125.

The changes in any replacement paragraph or section, or substitute specification must be shown by underlining (for added matter) or strikethrough (for deleted matter) with 2 exceptions: (1) for ***deletion of five characters or fewer, double brackets may be used (e.g., [[eroor]]); and (2) if strikethrough cannot be easily perceived (e.g., deletion of the number "4" or certain punctuation marks), double brackets must be used (e.g., [[4]]). As an alternative to using double brackets, however, extra portions of text may be included before and after text being deleted, all in strikethrough, followed by including and underlining the extra text with the desired change*** *(e.g., ~~number 4 as~~ number 14 as)*

C) Amendments to drawing figures:
Drawing changes must be made by presenting replacement figures which incorporate the desired changes and which comply with 37 CFR 1.84. An explanation of the changes made must be presented either in the drawing amendments, or remarks, section of the amendment, ***and may be accompanied by a marked-up copy of one or more of the figures being amended, with annotations.*** Any replacement drawing sheet must ***be identified in the top margin as "Replacement Sheet"*** and include all of the figures appearing on the immediate prior version of the sheet, even though only one figure may be amended. ***Any marked-up (annotated) copy showing changes must be labeled "Annotated Marked-up Drawings" and accompany the replacement sheet in the amendment (e.g., as an appendix).*** The figure or figure number of the amended drawing(s) must not be labeled as "amended." If the changes to the drawing figure(s) are not accepted by the examiner, applicant will be notified of any required corrective action in the next Office action. No further drawing submission will be required, unless applicant is notified.

Questions regarding the submission of amendments pursuant to the revised practice set forth in this flyer should be directed to: Elizabeth Dougherty or Gena Jones, Legal Advisors, or Joe Narcavage, Senior Special Projects Examiner, Office of Patent Legal Administration, by e-mail to patentpractice@uspto.gov or by phone at (703) 305-1616.

Notice of References Cited

Application/Control No.	Applicant(s)/Patent Under Reexamination	
10/728,551	PHILLIPS ET AL.	
Examiner	Art Unit	Page 1 of 1
Mark Connolly	2115	

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-6,996,709	02-2006	Arnold et al.	716/16
*	B	US-6,836,842	12-2004	Guccione et al.	713/100
*	C	US-6,571,381 B1	05-2003	Vorbach et al.	713/100
*	D	US-6,091,263	07-2000	New et al.	326/40
*	E	US-6,077,315 A	06-2000	Greenbaum et al.	717/157
*	F	US-6,021,490 A	02-2000	Vorbach et al.	713/100
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	Hennessy John L. et al, Computer Organization and Design: THE HARDWARE/SOFTWARE INTERFACE, 1998, Morgan Kaufmann Publishers, 2nd Ed., pgs. 545-549.
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

SECOND EDITION

Computer Organization and Design

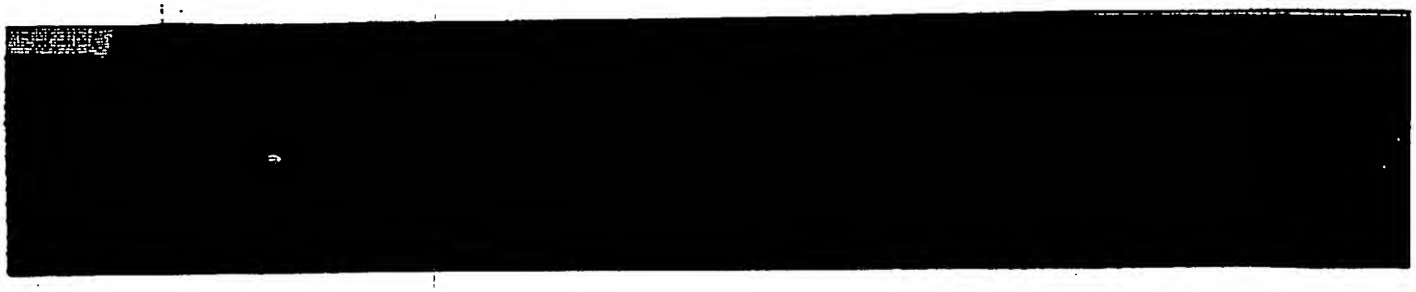
THE HARDWARE/SOFTWARE INTERFACE

John L. Hennessy
Stanford University

David A. Patterson
University of California, Berkeley

With a contribution by
James R. Larus
University of Wisconsin

Morgan Kaufmann Publishers, Inc.
San Francisco, California



Sponsoring Editor Denise Penrose
Production Manager Yonie Overton
Production Editor Julie Pabst
Editorial Coordinator Jane Elliott
Text and Cover Design Ross Carron Design
Illustration Alexander Teshin Associates, with second edition modifications by Dartmouth Publishing, Inc.
Chapter Opener Illustrations Canary Studios
Copyeditor Ken DellaPenta
Composition Nancy Logan
Proofreader Jennifer McClain
Indexer Steve Rath
Printer Courier Corporation

Morgan Kaufmann Publishers, Inc.
Editorial and Sales Office:
340 Pine Street, Sixth Floor
San Francisco, CA 94104-3205
USA

Telephone 415/392-2665
Facsimile 415/982-2665
Email *mkp@mkp.com*
WWW *http://www.mkp.com*
Order toll free 800/745-7323

Printed in the United States of America

02 01 00 99 5 4 3

Advice, Praise, and Errors: Any correspondence related to this publication or intended for the authors should be sent electronically to *cod2bugs@mkp.com*. Information regarding error sightings is encouraged. Any error sightings that are accepted for correction in subsequent printings will be rewarded by the authors with a payment of $1.00 (U.S.) per correction at the time of their implementation in a reprint.

Library of Congress Cataloging-in-Publication Data
Patterson, David A.
Computer organization and design : the hardware/software interface
/ David A. Patterson, John L. Hennessy.—2nd ed.
p. cm.
Includes bibliographical references and index.
ISBN 1-55860-428-6 (cloth).—ISBN 1-55860-491-X (paper)
1. Computer organization. 2. Computers—Design and construction.
3. Computer interfaces. I. Hennessy, John L. II. Title
QA76.9.C643H46 1997
004.2'2—dc21 97-16050

7.2 The Basics of Caches

Cache: a safe place for hiding or storing things.

Webster's New World Dictionary of the American Language, Third College Edition (1988)

In our library example, the desk acted as a cache—a safe place to store things (books) that we needed to examine. *Cache* was the name chosen to represent the level of the memory hierarchy between the CPU and main memory in the first commercial machine to have this extra level. Today, although this remains the dominant use of the word *cache*, the term is also used to refer to any storage managed to take advantage of locality of access. Caches first appeared in research machines in the early 1960s and in production machines later in that same decade; virtually every general-purpose machine built today, from the fastest to the slowest, includes a cache.

In this section, we begin by looking at a very simple cache in which the processor requests are each one word and the blocks also consist of a single word. Figure 7.4 shows such a simple cache, before and after requesting a data item that is not initially in the cache. Before the request, the cache contains a collection of recent references X1, X2, . . . , X*n* – 1, and the processor requests a word X*n* that is not in the cache. This request results in a miss, and the word X*n* is brought from memory into cache.

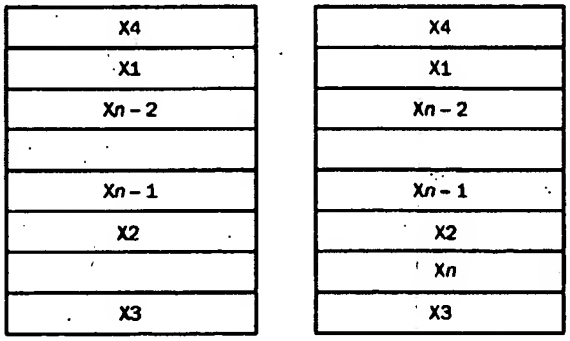


FIGURE 7.4 The cache just before and just after a reference to a word X*n* that is not initially in the cache. This reference causes a miss that forces the cache to fetch X*n* from memory and insert it into the cache.

Looking at the scenario in Figure 7.4, we can see that there are two questions we must answer: How do we know if a data item is in the cache? And, if it is, how do we find it? The answers to these two questions are related. If each word can go in exactly one place in the cache, then we will know how to find the word if it is in the cache. The simplest way to assign a location in the cache for each word in memory is to assign the cache location based on the address of the word in memory. This cache structure is called *direct mapped*, since each memory location is mapped to exactly one location in the cache. The typical mapping between addresses and cache locations for a direct-mapped cache is usually simple. For example, almost all direct-mapped caches use the mapping:

(Block address) modulo (Number of cache blocks in the cache)

This mapping is attractive because if the number of entries in the cache is a power of two, then modulo can be computed simply by using only the low-order $\log_2$ (cache size in blocks) bits of the address; hence the cache may be accessed directly with the low-order bits. For example, Figure 7.5 shows a direct-mapped cache of eight words and the memory addresses between 1_{ten} (00001_{two}) and 29_{ten} (11101_{two}) that map to locations 1_{ten} (001_{two}) and 5_{ten} (101_{two}) in the cache.

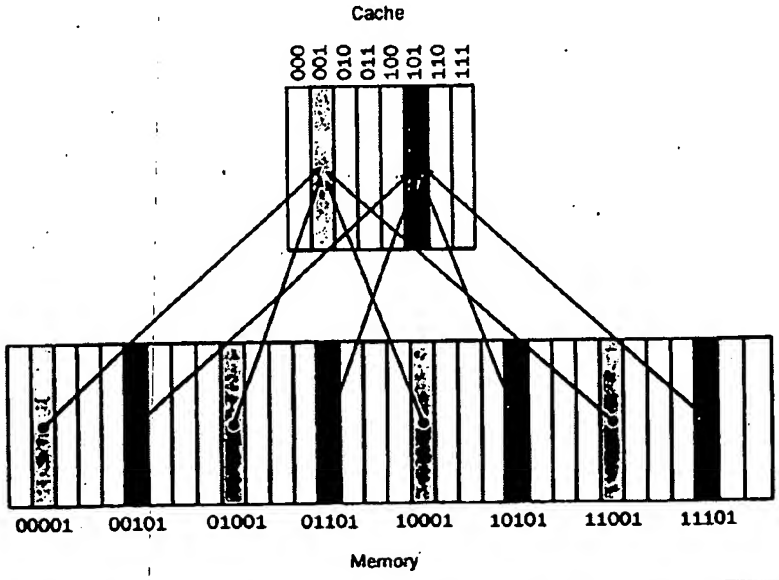


FIGURE 7.5 A direct-mapped cache with eight entries showing the addresses of memory words between 0 and 31 that map to the same cache locations. Because there are eight words in the cache, an address X maps to the cache word X modulo 8. That is, the low-order $\log_2(8) = 3$ bits are used as the cache index. Thus, addresses 00001_{two}, 01001_{two}, 10001_{two}, and 11001_{two} all map to entry 001_{two} of the cache, while addresses 00101_{two}, 01101_{two}, 10101_{two}, and

Because each cache location can contain the contents of a number of different memory locations, how do we know whether the data in the cache corresponds to a requested word? That is, how do we know whether a requested word is in the cache or not? We can plan for this by adding a set of *tags* to the cache. The tags contain the address information required to identify whether a word in the cache corresponds to the requested word. The tag needs only to contain the upper portion of the address, corresponding to the bits that are not used as an index into the cache. The reason for this is that the bits corresponding to the index are used to select the unique entry in the cache corresponding to the supplied address, and thus we only need to ensure that the upper portion of the supplied address matches the tag. For example, in Figure 7.5 we need only have 2 of the 5 address bits in the tag, since the lowest 3 bits of the address select the block.

We also need a way to recognize that a cache block does not have valid information. For instance, when a processor starts up, the cache will be empty, and the tag fields will be meaningless. Even after executing many instructions, some of the cache entries may still be empty, as in Figure 7.4. Thus we need to know that the tag should be ignored for such entries. The most common method is to add a *valid bit* to indicate whether an entry contains a valid address. If the bit is not set, there cannot be a match for this block.

For the rest of this section, we will focus on explaining how reads work in a cache and how the cache control works for reads. In general, handling reads is a little simpler than handling writes, since reads do not have to change the contents of the cache. After seeing the basics of how reads work and how cache misses can be handled, we'll examine the cache designs for two real machines and detail how these caches handle writes.

Accessing a Cache

Figure 7.6 shows the contents of an eight-word direct-mapped cache as it responds to a series of requests from the processor. Since there are eight blocks in the cache, the low-order 3 bits of an address give the block number. Here is the action for each reference:

Decimal address of reference	Binary address of reference	Hit or miss in cache	Assigned cache block (where found or placed)
22	10110_{two}	miss (7.5b)	$(10110_{two} \bmod 8) = 110_{two}$
26	11010_{two}	miss (7.5c)	$(11010_{two} \bmod 8) = 010_{two}$
22	10110_{two}	hit	$(10110_{two} \bmod 8) = 110_{two}$
26	11010_{two}	hit	$(11010_{two} \bmod 8) = 010_{two}$
16	10000_{two}	miss (7.5d)	$(10000_{two} \bmod 8) = 000_{two}$
3	00011_{two}	miss (7.5e)	$(00011_{two} \bmod 8) = 011_{two}$
16	10000_{two}	hit	$(10000_{two} \bmod 8) = 000_{two}$
18	10010_{two}	miss (7.5f)	$(10010_{two} \bmod 8) = 010_{two}$

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	N		
111	N		

a. The initial state of the cache after power-on

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	Y	10_{two}	Memory(10110_{two})
111	N		

b. After handling a miss of address (10110_{two})

Index	V	Tag	Data
000	N		
001	N		
010	Y	11_{two}	Memory (11010_{two})
011	N		
100	N		
101	N		
110	Y	10_{two}	Memory (10110_{two})
111	N		

c. After handling a miss of address (11010_{two})

Index	V	Tag	Data
000	Y	10_{two}	Memory (10000_{two})
001	N		
010	Y	11_{two}	Memory (11010_{two})
011	N		
100	N		
101	N		
110	Y	10_{two}	Memory (10110_{two})
111	N		

d. After handling a miss of address (10000_{two})

Index	V	Tag	Data
000	Y	10_{two}	Memory (10000_{two})
001	N		
010	Y	11_{two}	Memory (11010_{two})
011	Y	00_{two}	Memory (00011_{two})
100	N		
101	N		
110	Y	10_{two}	Memory (10110_{two})
111	N		

e. After handling a miss of address (00011_{two})

Index	V	Tag	Data
000	Y	10_{two}	Memory (10000_{two})
001	N		
010	Y	10_{two}	Memory (10010_{two})
011	Y	00_{two}	Memory (00011_{two})
100	N		
101	N		
110	Y	10_{two}	Memory (10110_{two})
111	N		

f. After handling a miss of address (10010_{two})

FIGURE 7.6 The cache contents are shown after each reference request that *misses*, with the index and tag fields shown in binary. The cache is initially empty, with all valid bits (V entry in cache) turned off (N). The processor requests the following addresses: 10110_{two} (miss), 11010_{two} (miss), 10110_{two} (hit), 11010_{two} (hit), 10000_{two} (miss), 00011_{two} (miss), 10000_{two} (hit), and 10010_{two} (miss). The figures show the cache contents after each miss in the sequence has been handled. When address 10010_{two} (18) is referenced, the entry for address 11010_{two} (26) must be replaced, and a reference to 11010_{two} will cause a subsequent miss. The tag field will contain only the upper portion of the address. The full address of a word contained in cache block i with tag field j for this cache is $j \times 8 + i$, or equivalently the concatenation of the tag field j and the index i. For example, in cache f above, index 010 has tag 10 and corresponds to address 10010.

When the word at address 18 (10010_{two}) is brought into cache block 2 (010_{two}), the word at address 26 (11010_{two}), which was in cache block 2 (010_{two}), must be replaced by the newly requested data. This behavior allows a cache to take advantage of temporal locality: recently accessed words replace less recently

referenced words. This situation is directly analogous to needing a book from the shelves and having no more space on your desk—some book already on your desk must be returned to the shelves. In a direct-mapped cache, there is only one place to put the newly requested item and hence only one choice of what to replace.

We know where to look in the cache for each possible address: the low-order bits of an address can be used to find the unique cache entry to which the address could map. Figure 7.7 shows how a referenced address is divided into

- a cache index, which is used to select the block
- a tag field, which is used to compare with the value of the tag field of the cache

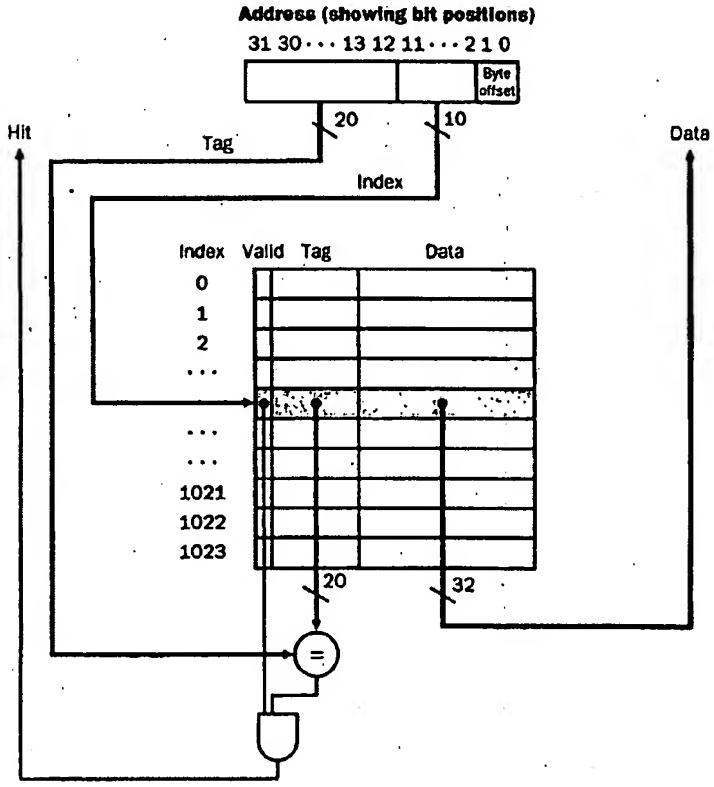


FIGURE 7.7 For this cache, the lower portion of the address is used to select a cache entry consisting of a data word and a tag. The tag from the cache is compared against the upper portion of the address to determine whether the entry in the cache corresponds to the requested address. Because the cache has 2^{10} (or 1024) words, and a block size of 1 word, 10 bits are used to index the cache, leaving 32 – 10 – 2 = 20 bits to be compared against the tag. If the tag and upper 20 bits of the address are equal and the valid bit is on, then the request hits in the cache, and the word is supplied to the processor. Otherwise, a miss occurs.